Azure PowerShell

## **Azure PowerShell**

Azure PowerShell is designed for managing and administering Azure resources from the command line, and for building automation scripts that work against the Azure Resource Manager. PowerShell Modules are used for automation and some settings are only available in PowerShell and not on portal. There are different ways to install Azure PowerShell Module.

* [Microsoft Web Platform Installer](https://www.microsoft.com/web/handlers/webpi.ashx/getinstaller/WindowsAzurePowershellGet.3f.3f.3fnew.appids)
* [PowerShell Gallery](https://www.powershellgallery.com/)

Installing Azure PowerShell from the PowerShell Gallery is the preferred method of installation.

**1. Install PowerShellGet**

Installing items from the PowerShell Gallery requires the PowerShellGet module

| OS Version | Install instructions |
| --- | --- |
| Windows 10 or Windows Server 2016 | Built into Windows Management Framework (WMF) 5.0 included in the OS |
| To upgrade to PowerShell 5 | [Install the latest version of WMF](https://www.microsoft.com/en-us/download/details.aspx?id=54616) |
| Windows with PowerShell 3 or PowerShell 4 | [Get the PackageManagement modules](http://go.microsoft.com/fwlink/?LinkID=746217) |

**2. Set Execution Policy**

Open Windows PowerShell command and run following commands.

The Start-Transcript cmdlet creates a record of all or part of a Windows PowerShell session to a text file. The transcript includes all command that the user types and all output that appears on the console.

The Set-ExecutionPolicy cmdlet enables you to determine which Windows PowerShell scripts (if any) will be allowed to run on your computer. Windows PowerShell has four different execution policies:

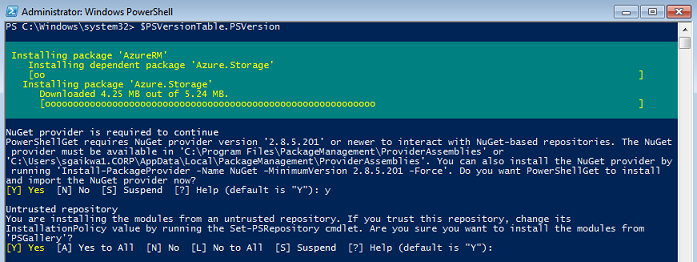
Restricted– No scripts can be run. Windows PowerShell can be used only in interactive mode.  
AllSigned– Only scripts signed by a trusted publisher can be run.  
RemoteSigned– Downloaded scripts must be signed by a trusted publisher before they can be run.  
Unrestricted– No restrictions; all Windows PowerShell scripts can be run.

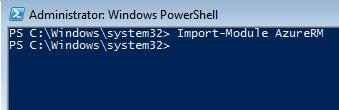


**3. Install Azure PowerShell**  
Installing Azure PowerShell from the PowerShell Gallery requires elevated privileges. Run the following command from an elevated PowerShell session:

Install-Module AzureRM

By default, the PowerShell gallery is not configured as a trusted repository for PowerShellGet. The first time you use the PSGallery you see the prompt of untrusted repository, you need to enter Y to install modules from PSGallery.



Once the module is installed, need to load the module into PowerShell session.

Import-Module AzureRm

**4. Login to Azure**  
To use any of the cmdlets in the AzureRM module, you need to log in. This requires that you have an Azure subscription.



Login-AzureRmAccount

5. Get Credentials for VM Admin  
Get credentials and store in variable, these credentials with be used as a admin user for VM.

$cred = Get-Credential

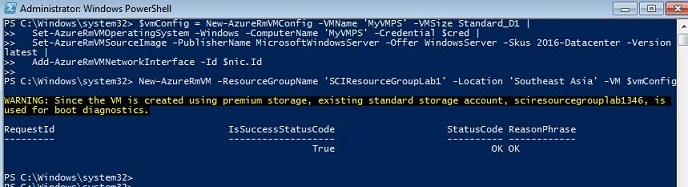
**6. Create New Azure VM**

I am using existing resource group and NIC information.

Get existing nic details from existing resource group  
$nic = Get-AzureRmNetworkInterface -Name myvm478 -ResourceGroupName  ‘SCIResourceGroupLab1’

Prepare configuration of VM, following is the single command to store VMconfiguration in variable vmConfig.  
$vmConfig = New-AzureRmVMConfig -VMName ‘MyVMPS’ -VMSize Standard\_D1 |  
Set-AzureRmVMOperatingSystem -Windows -ComputerName ‘MyVMPS’ -Credential $cred |  
Set-AzureRmVMSourceImage -PublisherName MicrosoftWindowsServer -Offer WindowsServer -Skus 2016-Datacenter -Version latest |  
Add-AzureRmVMNetworkInterface -Id $nic.Id

Create New Azure VM using vmConfig and resource group  
New-AzureRmVM -ResourceGroupName ‘SCIResourceGroupLab1’ -Location ‘Southeast Asia’ -VM $vmConfig



you can see MyVMPS in Azure portal after creation.